

Tic Tac Toe

Themen

Mit dieser Aufgabe wollen wir Folgendes trainieren:

- Objektorientierung
- Klassen, Instanzvariablen, Methoden

Beschreibung

Die `TicTacToe`-Klasse repräsentiert ein 3x3-Spielfeld für das bekannte gleichnamige Spiel. Die Klasse verwaltet den Spielstand und kann das Spielfeld mit den enthaltenen Zügen ausgeben.

Der Zustand einer Zelle wird mit einem Integer-Wert wie folgt codiert:

- 0 bedeutet, dass die Zelle leer ist.
- 1 bedeutet, dass die Zelle durch ein 'X' belegt ist.
- 2 bedeutet, dass die Zelle durch ein 'O' belegt ist.

Die Klasse besitzt folgende Methoden:

- Die `reset()`-Methode setzt das Spielfeld auf den Anfangszustand (alle Zellen sind leer). Diese Methode sollen wir auch im Konstruktor zur Initialisierung des gesamten Spielfelds verwenden.
- Die `setMark()`-Methode setzt das Spielerzeichen ('X' oder 'O') auf die angegebene Position. Liegt die Position ausserhalb des Spielfelds oder ist die damit angegebene Zelle bereits belegt, gibt die Methode `false` zurück. Anderenfalls wird das Spielerzeichen gesetzt.

Achtung: Welches Spielerzeichen zu setzen ist, verwaltet die `TicTacToe`-Klasse intern.

- Die `toString()`-Methode gibt das Spielfeld als String zurück. Dabei wollen wir den folgenden Aufbau implementieren (leeres Spielfeld):

```
| |  
-+-+--  
| |  
-+-+--  
| |
```

Aufgabenstellung

Schreibe die `TicTacToe`-Klasse mit den beschriebenen Funktionalitäten. Das Spielfeld soll dabei in Form eines eindimensionalen Integer-Arrays intern gespeichert werden.

Testfälle

Dieser Beispieldurchlauf soll die Funktion noch einmal verdeutlichen:

- Initialisierung

```
| |  
-+-+--  
| |  
-+-+--  
| |
```

■ `setMark(2,2)`

```
| |  
-+-+  
| |  
-+-+  
| |X
```

■ `setMark(2,0)`

```
| |O  
-+-+  
| |  
-+-+  
| |X
```

■ `setMark(1,1)`

```
| |O  
-+-+  
|X|  
-+-+  
| |X
```

Algorithmische Tipps

Wenn du stockst und nicht weiterweissst, dann versuch mal Folgendes:

- Du könntest das Spielfeld so speichern, dass nach drei Elementen jeweils die nächste Zeile folgt.
- Sorge dafür, dass die `setMark()` nur gültige Spielfeldgrössen akzeptiert. Wenn dies der Fall ist, kannst du die Position im Array bestimmen, da du schliesslich weisst, wie viele Elemente jede Zeile und Spalte im Spielfeld hat. Sobald du die Markierung gesetzt hast, kannst du die nächste Markierung bestimmen (zum Beispiel mit dem Modulo-Operator).
- Beim Zeichnen des Spielfeldes kannst du jedes Element einzeln durchgehen und das entsprechende Zeichen ('O', 'X' oder Leerzeichen) in die Konsole schreiben. Mit dem Modulo-Operator kannst du anschliessend prüfen, ob das aktuelle Element das letzte Element in der Zeile ist. Wenn dies der Fall ist, kannst du die neue Zeile malen und ansonsten das Zeichen für die Spaltenmarkierung.